UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/800,493	03/15/2004	Stephen Fife Sheldon	11466	9307

26890 7590 08/20/2007

JAMES M. STOVER
NCR CORPORATION
1700 SOUTH PATTERSON BLVD, WHQ3
DAYTON, OH 45479

EXAMINER	
SANDERS, AARON J	

ART UNIT	PAPER NUMBER
2168	

MAIL DATE	DELIVERY MODE
08/20/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

MN

Office Action Summary	Application No.	Applicant(s)	
	10/800,493	SHELDON ET AL.	
	Examiner	Art Unit	
	Aaron Sanders	2168	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *20 April 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-42* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-42* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *15 March 2004* is/are: a)☐ accepted or b)☒ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Continued Examination

As per the panel decision from the pre-appeal brief review, prosecution of the instant application has been reopened. Claims 1-42 are pending.

Drawings

The drawings are objected to under 37 CFR 1.83(a). The drawings must show every feature of the invention specified in the claims. Therefore, the method of claims 1-14 and the system of claims 29-42 must be shown or the features canceled from the claims. No new matter should be entered.

Corrected drawing sheets in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application. Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. The figure or figure number of an amended drawing should not be labeled as "amended." If a drawing figure is to be canceled, the appropriate figure must be removed from the replacement sheet, and where necessary, the remaining figures must be renumbered and appropriate changes made to the brief description of the several views of the drawings for consistency. Additional replacement sheets may be necessary to show the renumbering of the remaining figures. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either "Replacement Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d). If the changes are not accepted by the examiner, the applicant will

be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Specification

The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed. The following title is suggested: Performing SQL Query Optimization by Simplifying Sub-Expressions.

The use of at least the trademarks TERADATA ACTIVE DATA WAREHOUSING SYSTEM and NCR CORPORATION have been noted in this application. They should be capitalized wherever they appear and be accompanied by the generic terminology. Although the use of trademarks is permissible in patent applications, the proprietary nature of the marks should be respected and every effort made to prevent their use in any manner which might adversely affect their validity as trademarks.

As per paragraph [0003], the last phrase, "and selecting an optimal plan from executing the database query" is incorrect. It appears that it should read, "and selecting an optimal plan for executing the database query". Applicant should check the specification for other such errors.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 1-42 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

The disclosed subject matter lacks a practical application of a judicial exception (law of nature, abstract idea, naturally occurring article/phenomena) since it fails to produce a useful, concrete, and tangible result.

Specifically, the disclosed subject matter does not produce a tangible result because it fails to produce a result that is limited to having real world value rather than a result that may be interpreted to be abstract in nature as, for example, a thought, a computation, or manipulation of data. More specifically, the disclosed subject matter provides for performing query optimization and selecting an optimal plan for executing the database query. This produced result remains in the abstract because the result is not displayed to a user or output to another system and thus, fails to achieve the required status of having real world value.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1-3, 6-9, 11-17, 20-23, 25-31, 34-37, and 39-42 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. 6,665,664 (hereinafter Paulley).

1. A method of processing a database query, the query including one or more expressions, the method including:

performing expression optimization on one or more of the expressions (See e.g. col. 7, lines 42-55, "a preprocessing phase, in which expressions are simplified whenever possible");

performing further query optimization (See e.g. col. 7, lines 42-55, "a normalization phase, in which the simplified expression is analyzed and either fully converted to conjunctive normal form");

where the expression optimization is performed before further query optimization (See e.g. col. 7, lines 42-55, "This preprocessing phase includes several steps that are designed to simplify the original query expression, thereby simplifying the matrix processing occurring in the normalization phase"); and

where each expression includes one or more sub-expressions, and where the expression optimization includes (See e.g. col. 13, lines 1-13, "The present invention repeatedly generates prime implicates of disjunctive sub-expressions nested within a conjunctive expression, thereby normalizing the search condition piece-by-piece"), for each expression:

(1) if the expression has a form selected from the group consisting of "SE+0," "SE*1," and "SE/1," where SE is a sub-expression, then reducing the expression to SE (Paulley does not explicitly teach reducing expressions such as SE+0 to SE. However, Paulley does teach eliminating tautologies and numerous other methods of simplifying sub-expressions, see col. 13, lines 30-64. Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to reduce expressions of the form SE+0 to SE because such a simplification would have allowed Paulley's method and system to gain lower cost in executing queries on a database, see col. 2, lines 37-54).

2. The method of claim 1, where each expression includes one or more sub-expressions, and where the expression optimization includes, for each expression:

(2) if the expression has a form selected from the group consisting of "SE*0," "0/SE," and "0 MOD SE," where SE is a non-nullable sub-expression, then reducing the expression to 0 (Paulley does not explicitly teach reducing expressions such as SE*0 to 0. However, Paulley does teach eliminating tautologies and numerous other methods of simplifying sub-expressions, see col. 13, lines 30-64. Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to reduce expressions of the form SE+0 to SE because such a simplification would have allowed Paulley's method and system to gain lower cost in executing queries on a database, see col. 2, lines 37-54); and

(3) if the expression is of the form F(C), where F is a function and C is a constant and F(C) returns a return value, then reducing the expression to the return value (See e.g. col. 13, lines 50-64, "Step 403... simplifies the expression by... simplifying predicates and operator conditions as follows: (a) Folding constant expressions when the expressions contain integers (e.g., x=3+4 is Changed to x=7) and the columns referenced in the statement are numeric" where "x=3+4'' is a function in the form "F(C)").

3. The method of claim 2, where one or more of the sub-expressions include sub-expressions, the method including;

(4) for each sub-expression that includes a sub-expression, simplifying the sub-expression using (1)-(3) (See e.g. col. 13, lines 1-13, "The present invention repeatedly generates prime implicates of disjunctive sub-expressions nested within a conjunctive expression, thereby normalizing the search condition piece-by-piece").

6. The method of claim 1, where the query is represented by a tree, including one or more nodes (See e.g. col. 11, lines 56-65, "the SQL statements are passed to the parser 361 which converts the statements into a query tree").

7. The method of claim 1, where the query includes an assignment list clause and where one or more of the expressions are in the assignment list clause (See e.g. col. 18, lines 37-54, "A linked list of pointers is used to track which branches in the expression tree should be converted").

8. The method of claim 1, where the query includes a WHERE clause, and where one or more of the expressions are in the WHERE clause (See e.g. col. 2, Table 1, "SELECT name FROM employees WHERE sal=l 0,000").

9. The method of claim 1, where further query optimization includes:

determining a satisfiability of the database query (See e.g. col. 2, lines 55-63, "Conjunctive conditions are useful because they must each evaluate to true in order for the query's Where clause to be satisfied").

11. The method of claim 1, where further query optimization includes:

determining one or more plans for executing the query (See e.g. col. 2, lines 37-54, "a component called the optimizer determines the 'plan' or the best method of accessing the data to implement the SQL query").

12. The method of claim 11, where one of the one or more plans includes:

scanning a table to locate rows that satisfy one or more conditions; and

summing one or more columns in the rows that satisfy the one o1: more conditions (See e.g. col. 4, lines 11-43, "The usefulness of converting the search conditions to conjunctive

normal form is that for a clause that consists of only a single predicate (i.e., not 'ORed with anything'), for any row in the result of that query that predicate must be true" where then summing the columns would have the same "true" result).

13. The method of claim 1, where further query optimization includes:

selecting an optimal plan from executing the database query (See e.g. col. 12, lines 7-16, "The optimizer, therefore, performs an analysis of the query and picks the best execution plan, which in turn results in particular ones of the access methods being invoked during query execution").

15. A computer program, stored on a tangible storage medium, for use in processing a database query, the query including one or more expressions, the computer program including executable instructions that cause a computer to (See e.g. Fig. 2):

perform expression optimization on one or more of the expressions (See e.g. col. 7, lines 42-55, "a preprocessing phase, in which expressions are simplified whenever possible");

perform further query optimization (See e.g. col. 7, lines 42-55, "a normalization phase, in which the simplified expression is analyzed and either fully converted to conjunctive normal form");

where the expression optimization is performed before further query optimization (See e.g. col. 7, lines 42-55, "This preprocessing phase includes several steps that are designed to simplify the original query expression, thereby simplifying the matrix processing occurring in the normalization phase"), and where the computer program including executable instructions that cause a computer to, for each expression (See e.g. col. 13, lines 1-13, "The present invention

repeatedly generates prime implicates of disjunctive sub-expressions nested within a Conjunctive expression, thereby normalizing the search condition piece-by-piece"):

(1) determine if the expression has a form selected from the group consisting of "SE+0," "SE*1," and "SE/1," where SE is a sub-expression, and if so, then reduce the expression to SE (Paulley does not explicitly teach reducing expressions such as SE+0 to SE. However, Paulley does teach eliminating tautologies and numerous other methods of simplifying sub-expressions, see col. 13, lines 30-64. Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to reduce expressions of the form SE+0 to SE because such a simplification would have allowed Paulley's method and system to gain lower cost in executing queries on a database, see col. 2, lines 37-54).

16. The computer program of claim 15, where each expression includes one or more sub-expressions, and where the expression optimization includes, for each expression:

(2) if the expression has a form selected from the group consisting of "SE*0," "0/SE," and "0 MOD SE," where SE is a non-nullable sub-expression, then reducing the expression to 0 (Paulley does not explicitly teach reducing expressions such as SE*0 to 0. However, Paulley does teach eliminating tautologies and numerous other methods of simplifying sub-expressions, see col. 13, lines 30-64. Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to reduce expressions of the form SE+0 to SE because such a simplification would have allowed Paulley's method and system to gain lower cost in executing queries on a database, see col. 2, lines 37-54); and

(3) if the expression is of the form F(C), where F is a function and C is a constant and F(C) returns a return value, then reducing the expression to the return value (See e.g. col. 13, lines 50-64, "Step 403... simplifies the expression by... simplifying predicates and operator conditions as follows: (a) Folding constant expressions when the expressions contain integers (e.g., x=3+4 is Changed to x=7) and the columns referenced in the statement are numeric" where "x=3+4'' is a function in the form "F(C)").

17. The computer program of claim 16, where one or more of the sub-expressions include sub-expressions, the method including;

(4) for each sub-expression that includes a sub-expression, simplifying the sub-expression using (1)-(3) (See e.g. col. 13, lines 1-13, "The present invention repeatedly generates prime implicates of disjunctive sub-expressions nested within a conjunctive expression, thereby normalizing the search condition piece-by-piece").

20. The computer program of claim 15, where the query is represented by a tree, including one or more nodes (See e.g. col. 11, lines 56-65, "the SQL statements are passed to the parser 361 which converts the statements into a query tree").

21. The computer program of claim 15, where the query includes an assignment list clause and where one or more of the expressions are in the assignment list clause (See e.g. col. 18, lines 37-54, "A linked list of pointers is used to track which branches in the expression tree should be converted").

22. The computer program of claim 15, where the query includes a WHERE clause, and where one or more of the expressions are in the WHERE clause (See e.g. col. 2, Table 1, "SELECT name FROM employees WHERE sal=l 0,000").

23. The computer program of claim 15, where further query optimization includes:

determining a satisfiability of the database query (See e.g. col. 2, lines 55-63, "Conjunctive conditions are useful because they must each evaluate to true in order for the query's Where clause to be satisfied").

25. The computer program of claim 15, where further query optimization includes:

determining one or more plans for executing the query (See e.g. col. 2, lines 37-54, "a component called the optimizer determines the 'plan' or the best method of accessing the data to implement the SQL query").

26. The computer program of claim 25, where one of the one or more plans includes:

scanning a table to locate rows that satisfy one or more conditions; and

summing one or more columns in the rows that satisfy the one o1: more conditions (See e.g. col. 4, lines 11-43, "The usefulness of converting the search conditions to conjunctive normal form is that for a clause that consists of only a single predicate (i.e., not 'ORed with anything'), for any row in the result of that query that predicate must be true" where then summing the columns would have the same "true" result).

27. The computer program of claim 15, where further query optimization includes:

selecting an optimal plan from executing the database query (See e.g. col. 12, lines 7-16, "The optimizer, therefore, performs an analysis of the query and picks the best execution plan, which in turn results in particular ones of the access methods being invoked during query execution").

29. A database system including:

a massively parallel processing system including: one or more nodes (See e.g. Fig. 3);

a plurality of CPUs, each of the one or more nodes providing access to one or more CPUs (See e.g. Fig. 3);

a plurality of data storage facilities each of the one or more CPUs providing access to one or more data storage facilities (See e.g. Fig. 3);

a process for execution on the massively parallel processing system for processing one or more database queries, each query including one or more expressions (See e.g. Fig. 3), the process including:

performing expression optimization on one or more of the expressions (See e.g. col. 7, lines 42-55, "a preprocessing phase, in which expressions are simplified whenever possible");

performing further query optimization (See e.g. col. 7, lines 42-55, "a normalization phase, in which the simplified expression is analyzed and either fully converted to conjunctive normal form");

where the expression optimization is performed before the further query optimization (See e.g. col. 7, lines 42-55, "This preprocessing phase includes several steps that are designed to simplify the original query expression, thereby simplifying the matrix processing occurring in the normalization phase"); and

where each expression includes one or more sub-expressions, and where the expression optimization includes, for each expression (See e.g. col. 13, lines 1-13, "The present invention repeatedly generates prime implicates of disjunctive sub-expressions nested within a conjunctive expression, thereby normalizing the search condition piece-by-piece"):

(1) if the expression has a form selected from the group consisting of "SE+0," "SE*1," and "SE/1," where SE is a sub-expression, then reducing the expression to SE (Paulley does not explicitly teach reducing expressions such as SE+0 to SE. However, Paulley does teach eliminating tautologies and numerous other methods of simplifying sub-expressions, see col. 13, lines 30-64. Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to reduce expressions of the form SE+0 to SE because such a simplification would have allowed Paulley's method and system to gain lower cost in executing queries on a database, see col. 2, lines 37-54).

30. The database system of claim 29, where each expression includes one or more sub-expressions, and where the expression optimization includes, for each expression:

(2) if the expression has a form selected from the group consisting of "SE*0," "0/SE," and "0 MOD SE," where SE is a non-nullable sub-expression, then reducing the expression to 0 (Paulley does not explicitly teach reducing expressions such as SE*0 to 0. However, Paulley does teach eliminating tautologies and numerous other methods of simplifying sub-expressions, see col. 13, lines 30-64. Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to reduce expressions of the form SE+0 to SE because such a simplification would have allowed Paulley's method and system to gain lower cost in executing queries on a database, see col. 2, lines 37-54); and

(3) if the expression is of the form F(C), where F is a function and C is a constant and F(C) returns a return value, then reducing the expression to the return value (See e.g. col. 13, lines 50-64, "Step 403... simplifies the expression by... simplifying predicates and operator

conditions as follows: (a) Folding constant expressions when the expressions contain integers (e.g., x=3+4 is Changed to x=7) and the columns referenced in the statement are numeric" where "x=3+4'' is a function in the form "F(C)").

31. The database system of claim 30, where one or more of the sub-expressions include sub-expressions, the method including;

(4) for each sub-expression that includes a sub-expression, simplifying the sub-expression using (1)-(3) (See e.g. col. 13, lines 1-13, "The present invention repeatedly generates prime implicates of disjunctive sub-expressions nested within a conjunctive expression, thereby normalizing the search condition piece-by-piece").

34. The database system of claim 29, where the query is represented by a tree, including one or more nodes (See e.g. col. 11, lines 56-65, "the SQL statements are passed to the parser 361 which converts the statements into a query tree").

35. The database system of claim 29, where the query includes an assignment list clause and where one or more of the expressions are in the assignment list clause (See e.g. col. 18, lines 37-54, "A linked list of pointers is used to track which branches in the expression tree should be converted").

36. The database system of claim 29, where the query includes a WHERE clause, and where one or more of the expressions are in the WHERE clause (See e.g. col. 2, Table 1, "SELECT name FROM employees WHERE sal=l 0,000").

37. The database system of claim 29, where further query optimization includes:

determining a satisfiability of the database query (See e.g. col. 2, lines 55-63, "Conjunctive conditions are useful because they must each evaluate to true in order for the query's Where clause to be satisfied").

39. The database system of claim 29, where further query optimization includes:

determining one or more plans for executing the query (See e.g. col. 2, lines 37-54, "a component called the optimizer determines the 'plan' or the best method of accessing the data to implement the SQL query").

40. The database system of claim 39, where one of the one or more plans includes:

scanning a table to locate rows that satisfy one or more conditions; and

summing one or more columns in the rows that satisfy the one or more conditions (See e.g. col. 4, lines 11-43, "The usefulness of converting the search conditions to conjunctive normal form is that for a clause that consists of only a single predicate (i.e., not 'ORed with anything'), for any row in the result of that query that predicate must be true" where then summing the columns would have the same "true" result).

41. The database system of claim 29, where further query optimization includes:

selecting an optimal plan from executing the database query (See e.g. col. 12, lines 7-16, "The optimizer, therefore, performs an analysis of the query and picks the best execution plan, which in turn results in particular ones of the access methods being invoked during query execution").

Claims 4, 18, and 32 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. 6,665,664 (hereinafter Paulley), in view of U.S. 5,590,324 (hereinafter Leung).

4. The method of claim 2, where SE is nullable if it includes a nullable column (Paulley does not teach making a sub-expression nullable if it includes a nullable column. However, Leung does, see col. 7, lines 56-64, "if a head expression simply consists of a column C, then the output column retains the nullability of column C from its input derived table or base table". Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Leung's teachings would have allowed Paulley's method and system to gain lower cost in executing queries on a database, see Leung col. 1, lines 62-67, "The exploitation of column nullability can mean a potentially huge saving in query execution time").

Claims 5, 19, and 33 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. 6,665,664 (hereinafter Paulley), in view of U.S. 6,618,719 (hereinafter Andrei).

5. The method of claim 2, where SE is nullable if it belongs to an inner table of an outer join (Paulley does not teach making a sub-expression nullable if it belongs to an inner table of an outer join. However, Andrei does, see col. 27, lines 44-47, "The ASE query engine's single outer join algorithm requires the inner table of the outer join to be also the inner table of the join, to substitute NULLs when no inner row qualifies for a given outer row" where, see col. 2, lines 57-59, "ASE" is an "Adaptive Server Enterprise" and rows are part of a "derived table--A table implemented as a stream of rows, representing the result of a relational operator". Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Andrei's teachings would have allowed Paulley's method and system to gain lower cost in executing queries on a database, see Leung

col. 1, lines 62-67, "The exploitation of column nullability can mean a potentially huge saving in query execution time").

Claims 10, 14, 24, 28, 38, and 42 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. 6,665,664 (hereinafter Paulley), in view of Esko Nuutila, "Transitive Closure", Helsinki University of Technology, 9 October 1995 (hereinafter Nuutila).

10. The method of claim 1, where further query optimization includes:

determining a transitive closure of the database query (Paulley does not teach determining a transitive closure of the database query. However, Nuutila does: "The transitive closure of G is a graph G+ = (V,E+) such that for all v,w in V there is an edge (v,w) in E+ if and only if there is a non-null path from v to w in G". Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Nuutila's teachings would have allowed Paulley's method and system to gain lower cost in executing queries on a database, see Nuutila, "It is required, for instance, in the reachability analysis of transition networks representing distributed and parallel systems and in the construction of parsing automata in compiler construction").

14. The method of claim l, where further query optimization includes two or more optimizations selected from the group consisting of:

determining a satisfiability of the database query (See e.g. Paulley col. 2, lines 55-63, "Conjunctive conditions are useful because they must each evaluate to true in order for the query's Where clause to be satisfied");

determining a transitive closure of the database query (Paulley does not teach determining a transitive closure of the database query. However, Nuutila does: "The transitive closure of G is a graph G+ = (V,E+) such that for all v,w in V there is an edge (v,w) in E+ if and only if there is a non-null path from v to w in G". Thus, it would have been obvious to one of ordinary skill in the database art at the time of the invention to combine the teachings of the cited references because Nuutila's teachings would have allowed Paulley's method and system to gain lower cost in executing queries on a database, see Nuutila, "It is required, for instance, in the reachability analysis of transition networks representing distributed and parallel systems and in the construction of parsing automata in compiler construction");

determining one or more plans for executing the query (See e.g. Paulley col. 2, lines 37-54, "a component called the optimizer determines the 'plan' or the best method of accessing the data to implement the SQL query"); and

selecting an optimal plan from executing the database query (See e.g. Paulley col. 12, lines 7-16, "The optimizer, therefore, performs an analysis of the query and picks the best execution plan, which in turn results in particular ones of the access methods being invoked during query execution").

Response to Arguments

The Examiner has accepted the proposed title amendment, however, Applicant must submit the amended title in the response to the Office action.

As per Applicant's first argument that Paulley does not teach reducing expressions in the form SE+0, SE*1, and SE/1, the Examiner agrees that Paulley does not make such an explicit

teaching. Paulley teaches "eliminating tautologies". However, given the optimizations that Paulley does perform on expressions and an elementary knowledge of basic mathematics (e.g. the properties of 1 and 0), it would be obvious to one of ordinary skill in the art to reduce the expressions SE+0, SE*1, and SE/1 to SE.

As per Applicant's second argument that the claim language does not recite "SE*1 = SE" and is therefore not a tautology, the Examiner respectfully disagrees. According to the *American Heritage Dictionary of the English Language, Fourth Edition*, Houghton Mifflin Company, 2004, "reduce" means to "simplify the form of (an expression, such as a fraction) without changing the value" and "equal" means "having the same quantity, measure, or value as another". In this sense, the two are clearly equivalent, thus the limitation "reducing" SE*1 to SE is equivalent to "SE*1 = SE". Further, "SE*1 = SE" is a tautology because even if SE is false, a statement such as "False = False" is always true.

As per Applicant's third argument that reducing SE*1 to SE is not disclosed by Paulley when he describes "folding constant expressions", the Examiner agrees.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Aaron Sanders whose telephone number is 571-270-1016. The examiner can normally be reached on M-Th 8:00a-5:00p.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tim Vo can be reached on 571-272-3642. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/AJS/
Aaron J. Sanders
Examiner
9 August 2007

SRP
8/16

TIM VO
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100